

Scripts Definitions

19mon.sql

- a) Describe the file purpose - finds the people who have not logged on for 19 months
- b) What are the file inputs - reads user detail table
- c) What are the file outputs - writes to the emailist table
- d) How is the file called - called from oldID.sh
- e) What does the file call - This file does not call anything.
- f) In what order are the files called - look at the cron when oldID.sh is executed.

CODE

```
set long 5000;
Declare
cursor a1 is
Select
    a.USERNAME, b.FIRSTNAME, b.LASTNAME, b.EMAIL
FROM    userdetail a,users b
WHERE   a.lastlogin <= add_months(sysdate,-19) and
        a.username = b.username;

BEGIN
    FOR a1_rec in a1 LOOP
        insert into emailist (emailid,subject,message,username)
            VALUES (emailist_seq.nextval,'IFAP User Expiration Notice',
'Dear Partner:
```

You previously requested and received an ID for accessing the IFAP web sites "Member Services Area" (<<http://ifap.ed.gov>>). Our records indicate that it has been more than eighteen (18) months since you last accessed that area of the IFAP web site.

As part of IFAPs ID maintenance procedures, we will delete IFAP IDs after nineteen (19) months of inactivity. This email serves to notify you. that your ID will be deleted, unless you take one of the following actions.

To keep your IFAP Member Services ID active, please login to the Member Services area in IFAP. This will re-establish your use of the site. In addition, please go to the "Update User Profile" link on the IFAP Member Services Menu area to review your user profile for accuracy.

If you do not wish to keep your IFAP Member Services ID active, you do not need to do anything. We will delete your ID as a part of our regular maintenance.

Thank you.

```
SFA Customer Service Call Center
IFAP/Portal Website Manager',
    a1_rec.username);
```

```
END LOOP;

END;
```

20mon.sql

- a) Describe the file purpose - finds the people who have not logged on for 20 months
- b) What are the file inputs - reads user detail table
- c) What are the file outputs - writes to the emailist table. The RunReadEmail script reads this file and sends out emails.
- d) How is the file called - called from oldID.sh
- e) What does the file call - This file does not call anything.
- f) In what order are the files called - look at the cron to see when oldID.sh is executed.

CODE

```
set long 4500;
Declare
cursor a1 is
Select
        a.USERNAME, b.FIRSTNAME, b.LASTNAME, b.EMAIL
FROM      userdetail a,users b
WHERE     a.lastlogin <= add_months(sysdate,-20) and
        a.username = b.username;

BEGIN
        FOR a1_rec in a1 LOOP
                insert into emailist (emailid,subject,message,username)
                VALUES (emailist_seq.nextval,'IFAP User Deletion Notice',
'Dear Partner,

This message is to inform you that your IFAP Member Services ID '||
a1_rec.username||' has expired. Our records indicate that it has been
more than twenty (20) months since you last accessed that area of the
IFAP website. It is a part of our regular maintenance policy to delete
Member Services IDs after twenty (20) months of inactivity.

If you wish to continue the use of IFAPs Member Services, please re-
register under "Member Services" at (<http://ifap.ed.gov> ).
Thank you.

SFA Customer Service Call Center
IFAP/Portal Website Manager',
        a1_rec.username);

        END LOOP;
END;
```

oldID.sh

- a) Describe the file purpose - finds the people who have not logged on for 19 or 20 months
- b) What are the file inputs - The 19mon.sql and 20mon.sql are initiated from this shell script
- c) What are the file outputs - The two sql scripts initiated by this shell script write to the emailist table. The RunReadEmail script reads this file and sends out emails. If the script fails, a log file is created and an email is sent to the system administrator.
- d) How is the file called - called by the cron
- e) What does the file call - 19mon.sql and 20mon.sql
- f) In what order are the files called - 19mon.sql is called first followed by 20mon.sql

CODE

```
if [[ "$RANDOM" = "$RANDOM" ]]
then
    exec /bin/ksh $0 $*
fi

cat $EMAIL_HOME/CopyFiles | sqlplus ifapuser @$EMAIL_HOME/19mon.sql >
$EMAIL_HOME
E/Logs/19mon.Log1 2> $EMAIL_HOME/Logs/19mon.Log2

$EMAIL_HOME/ErrorHandler $EMAIL_HOME/Logs/19mon.Log1
$EMAIL_HOME/ErrorHandler $EMAIL_HOME/Logs/19mon.Log2
if [ $? = 0 ]
then
    continue
else
    echo 'Error in 19mon.sh' >> $EMAIL_HOME/sysmsg 2>&1
    exit
fi

cat $EMAIL_HOME/CopyFiles | sqlplus ifapuser @$EMAIL_HOME/20mon.sql >
$EMAIL_HOME
E/Logs/20mon.Log1 2> $EMAIL_HOME/Logs/20mon.Log2

$EMAIL_HOME/ErrorHandler $EMAIL_HOME/Logs/20mon.Log1
$EMAIL_HOME/ErrorHandler $EMAIL_HOME/Logs/20mon.Log2
if [ $? = 0 ]
then
    continue
else
    echo 'Error in 20mon.sh' >> $EMAIL_HOME/sysmsg 2>&1
    exit
fi
```

20mondel.sql

- a) Describe the file purpose - finds the people who have not logged on for 20 months and deleted all the information about these people from the database.
- b) What are the file inputs - reads the userdetail table
- c) What are the file outputs - There are no outputs. Records are deleted from the database.
- d) How is the file called - called by the oldIDdel.sh
- e) What does the file call - This file does not call anything.
- f) In what order are the files called - look at the cron for oldIDdel.sh.

CODE

```
Declare
cursor a1 is
Select
    a.USERNAME, b.FIRSTNAME, b.LASTNAME, b.EMAIL
FROM    userdetail a,users b
WHERE   a.lastlogin <= add_months(sysdate,-20) and
        a.username = b.username;

BEGIN
    FOR a1_rec in a1 LOOP
        DELETE from emailist where username = a1_rec.username;
        DELETE from storedquery where username = a1_rec.username;
        DELETE from subscriptions where username = a1_rec.username;
        DELETE from userdetail where username = a1_rec.username;
        DELETE from users where username = a1_rec.username;

        END LOOP;
END;
```

oldIDdel.sh

- a) Describe the file purpose - finds the people who have not logged on for 20 months
- b) What are the file inputs - The 20mondell.sql are initiated from this shell script
- c) What are the file outputs - The 20mondell.sql is initiated by this shell script. If the script fails, a log file is created and an email is sent to the system administrator.
- d) How is the file called - called by the cron
- e) What does the file call - 20mondell.sql
- f) In what order are the files called - 20mondell.sql is called first

CODE

```
if [[ "$RANDOM" = "$RANDOM" ]]
then
    exec /bin/ksh $0 $*
fi

cat $EMAIL_HOME/CopyFiles | sqlplus ifapuser @$EMAIL_HOME/20mondell.sql
> $EMAIL_
HOME/Logs/20mondell.Log1 2> $EMAIL_HOME/Logs/20mondell.Log2

$EMAIL_HOME/ErrorHandler $EMAIL_HOME/Logs/20mondell.Log1
$EMAIL_HOME/Logs/20mondell.
Log2
if [ $? = 0 ]
then
    continue
else
    echo 'Error in 20mondell.sh' >> $EMAIL_HOME/sysmsg 2>&1
    exit
fi
```

48hrsub.sql

- a) Describe the file purpose - This sql routine finds all of the users who want to be notified (every 48 hours) when a new document has been created.
- b) What are the file inputs - reads the subscription and ifapmetadata tables.
- c) What are the file outputs - Email Information is written to the emailist table.
- d) How is the file called - called by the 48hrsubscription.sh
- e) What does the file call - This file does not call anything.
- f) In what order are the files called - look at the cron for 48hrsubscription.sh.

CODE

Code is too long - See actual file

48hrsubscription.sh

- a) Describe the file purpose - finds all of the users who want to be notified (every 48 hours) when a new document has been created.
- b) What are the file inputs - The 48hrsub.sql is initiated from this shell script
- c) What are the file outputs - The 48hrsub.sql is initiated by this shell script. The sql script initiated by this shell script writes to the emailist table. The RunReadEmail script reads this file and sends out emails. If the script fails, a log file is created and an email is sent to the system administrator.
- d) How is the file called - called by the cron
- e) What does the file call - 48hrsub.sql
- f) In what order are the files called - 48hrsub.sql is called first

CODE

```
if [[ "$RANDOM" = "$RANDOM" ]]
then
    exec /bin/ksh $0 $*
fi

cat $EMAIL_HOME/CopyFiles | sqlplus ifapuser @$EMAIL_HOME/48hrsub.sql >
$EMAIL_H
OME/Logs/48hrsubscription.Log1 2>
$EMAIL_HOME/Logs/48hrsubscription.Log2

$EMAIL_HOME/ErrorMessage $EMAIL_HOME/Logs/48hrsubscription.Log1
$EMAIL_HOME/Logs/4
8hrsubscription.Log2
if [ $? = 0 ]
then
    continue
else
    echo 'Error in 48hrsubscription.sh' >> $EMAIL_HOME/sysmsg 2>&1
    exit
fi
```

citationfiles.sql

- a) Describe the file purpose - This sql routine extracts information from the database that is relevant to citations
- b) What are the file inputs - reads the citation and ifapmetadata tables.
- c) What are the file outputs - citation.txt, asofdfate.txt and updfed.txt.
- d) How is the file called - called by the citationfiles.sh
- e) What does the file call - This file does not call anything.
- f) In what order are the files called - look at the cron for citationfiles.sh.

CODE

```
set feedback off;
set termout off;
set heading off;
set pages 9999;
spool $EMAIL_HOME/updfed.txt;
select directoryandfilename||','||citations||asofdate
from ifapmetadata
where publicationtype = 'regcomps' and
trunc(asofdate) > add_months((trunc(sysdate , 'YEAR') - 1), -12)
order by citations, asofdate;
spool off;
spool $EMAIL_HOME/asofdate.txt;
select distinct asofdate||','||asofdate
from citation
where trunc(asofdate) > add_months((trunc(sysdate , 'YEAR') - 1), -12)
order by asofdate||','||asofdate;
spool off;
spool $EMAIL_HOME/citation.txt;
select
'(R)'||partnumber||'.'||sectionnumber||','||'(R)'||partnumber||'.'||sec
ti
onnumber||' '||asofdate
from citation
where trunc(asofdate) > add_months((trunc(sysdate , 'YEAR') - 1), -12)
order by partnumber||sectionnumber||subpartnumber||asofdate;
spool off;
```


citationfiles.sh

- a) Describe the file purpose - This shell script runs the citationfiles.sql. Once this sql script is completed, the citation.txt, asofdfate.txt and updfed.txt files are ftp'd to a directory on the 133 machine. Interwoven uses this information to populate some of its drop down boxes.
- b) What are the file inputs - The citationfiles.sql is initiated from this shell script
- c) What are the file outputs - citation.txt, asofdfate.txt and updfed.txt files are ftp'd to a directory on the 133 machine. If the script fails, a log file is created and an email is sent to the system administrator.
- d) How is the file called - called by the cron
- e) What does the file call - citationfiles.sql. It also initiates an FTP file transfer.
- f) In what order are the files called - citationfiles.sql is called first and then the ftp is initiated

CODE

```
# citationfiles.sh
# Description: Load the files required by Interwoven for the drop down
#              boxes on the templates. Once completed, ftp the files to
#              134 machine
if [[ "$RANDOM" = "$RANDOM" ]]
then
    exec /bin/ksh $0 $*
fi

cat $EMAIL_HOME/CopyFiles | sqlplus ifapuser
@$EMAIL_HOME/citationfiles.sql > $E
MAIL_HOME/Logs/citationfiles.Log1 2>
$EMAIL_HOME/Logs/citationfiles.Log2

$EMAIL_HOME/ErrorHandler $EMAIL_HOME/Logs/citationfiles.Log1
$EMAIL_HOME/Logs/cita
tionfiles.Log2
if [ $? = 0 ]
then
    continue
else
    echo 'Error in citationfiles.sh' >> $EMAIL_HOME/sysmsg 2>&1
    exit
fi

ftp -v -n 4.20.14.134 <<-!
user iw_cite id4ifap
pwd
cd /iw-home/local/config
lcd $EMAIL_HOME
put asofdate.txt
put citation.txt
put updfed.txt
quit
```

!

weeklysubscription.sql

- a) Describe the file purpose - This sql routine finds all of the users who want to be notified (weekly) when a new document has been created.
- b) What are the file inputs - reads the subscription and ifapmetadata tables.
- c) What are the file outputs - Email Information is written to the emailist table.
- d) How is the file called - called by the subscription.sh
- e) What does the file call - This file does not call anything.
- f) In what order are the files called - look at the cron for subscription.sh.

CODE

Code is too long - See actual file

subscription.sh

- a) Describe the file purpose - finds all of the users who want to be notified (weekly) when a new document has been created.
- b) What are the file inputs - The 48hrsub.sql is initiated from this shell script
- c) What are the file outputs - The weeklysubscription.sql is initiated by this shell script. The sql script initiated by this shell script writes to the emailist table. The RunReadEmail script reads this file and sends out emails. If the script fails, a log file is created and an email is sent to the system administrator.
- d) How is the file called - called by the cron
- e) What does the file call - weeklysubscription.sql
- f) In what order are the files called - weeklysubscription.sql is called first

CODE

```
# 134 machine
if [[ "$RANDOM" = "$RANDOM" ]]
then
    exec /bin/ksh $0 $*
fi

cat $EMAIL_HOME/CopyFiles | sqlplus ifapuser
@$EMAIL_HOME/weeklysubscription.sql
> $EMAIL_HOME/Logs/subscription.Log1 2>
$EMAIL_HOME/Logs/subscription.Log2

$EMAIL_HOME/ErrorCheck $EMAIL_HOME/Logs/subscription.Log1
$EMAIL_HOME/Logs/subsc
ription.Log2
if [ $? = 0 ]
then
    continue
else
    echo 'Error in subscription.sh' >> $EMAIL_HOME/sysmsg 2>&1
    exit
fi
```